

Tutorial 4: Triggers (*February 24, 2016*)

Flight tickets

```
Airports(_code_, city)
FlightCodes(_code_, airlineName)
Flights(origin, destination, departure, arrival, _code_)
  origin -> Airports.code
  destination -> Airports.code
  code -> FlightCodes.code
AvailableFlights(_flight_, _date_, numberOfFreeSeats, price)
  flight -> Flights.code
Bookings(_reference_, flight, date, passenger, price)
  (flight, date) -> AvailableFlights.(code, date)
```

```
Planes(_regnr_, capacity)
AvailableFlights(_flight_, _date_, numberOfFreeSeats, price, plane)
  (flight, date) -> Flights.(code, date)
  plane -> Planes.regnr
```

Restaurant

```
Tables(_number_, seats)
Bookings(_name_, _time_, nbpeople, table)
  table -> Tables.number
```

Wiki

```
PageRevision(_name_, _date_, author, text)
DeleteLog(_pagename_, _date_)
```

Trigger Cheat Sheet

```
CREATE OR REPLACE FUNCTION function_name() RETURNS TRIGGER AS $$
DECLARE [...]
BEGIN

    ...

    RETURN { NEW | OLD | NULL };
END
$$ LANGUAGE 'plpgsql';

CREATE TRIGGER name { BEFORE | AFTER | INSTEAD OF } { event [ OR ... ] }
ON table_name
[ FOR [ EACH ] { ROW | STATEMENT } ]
[ WHEN ( condition ) ]
EXECUTE PROCEDURE function_name ( arguments )
```

Where event can be one of:

```
INSERT
UPDATE [ OF column_name [, ... ] ]
DELETE
TRUNCATE
```

When	Event	FOR EACH ROW	FOR EACH STATEMENT
BEFORE/AFTER	INSERT/UPDATE/ DELETE	Tables	Tables/views
INSTEAD OF	INSERT/UPDATE/ DELETE	Views	–
